

**The Experiment Report of**

***Machine Learning***

**College Software College**

**Subject Software Engineering**

**Members**  Mo Junwen

**Student ID 201530612545**

**E-mail 545096186@qq.com**

**Tutor**   **Tan Mingkui**

**Date submitted** **2017. 12 .11**

**1. Topic:Logistic Regression, Linear Classification and Stochastic Gradient Descent**

**2. Time: 2017.12.11**

**3. Reporter:Mo Junwen**

**4. Purposes:**

(1)Compare and understand the difference between gradient descent and stochastic gradient descent.

(2)Compare and understand the differences and relationships between Logistic regression and linear classification.

(3)Further understand the principles of SVM and practice on larger data.

**5. Data sets and data analysis:**

Experiment uses a9a of LIBSVM Data, including 32561/16281(testing) samples and each sample has 123/123 (testing) features.

1. **Experimental steps:**

**Logistic Regression and Stochastic Gradient Descent**

(1)Load the training set and validation set.

(2)Initalize logistic regression model parameters, you can consider initalizing zeros, random numbers or normal distribution.

(3)Select the loss function and calculate its derivation, find more detail in PPT.

(4)Calculate gradient  toward loss function from partial samples.

(5)Update model parameters using different optimized methods(NAG，RMSProp，AdaDelta and Adam).

(6)Select the appropriate threshold, mark the sample whose predict scores greater than the threshold as positive, on the contrary as negative. (7)Predict under validation set and get the different optimized method loss LNAG，LRMSprop，LAdaDelta and LAdam .

(8)Repeat step 4 to 6 for several times, and drawing graph of LNAG，LRMSprop，LAdaDelta and LAdam  and with the number of iterations.

**Linear Classification and Stochastic Gradient Descent**

(1)Load the training set and validation set.

(2)Initalize SVM model parameters, you can consider initalizing zeros, random numbers or normal distribution.

(3)Select the loss function and calculate its derivation, find more detail in PPT.

(4)Calculate gradient  toward loss function from partial samples.

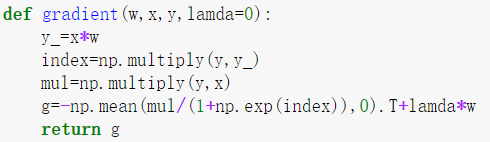
(5)Update model parameters using different optimized methods(NAG，RMSProp，AdaDelta and Adam).

(6)Select the appropriate threshold, mark the sample whose predict scores greater than the threshold as positive, on the contrary as negative. (7)Predict under validation set and get the different optimized method loss LNAG，LRMSprop，LAdaDelta and LAdam  and .

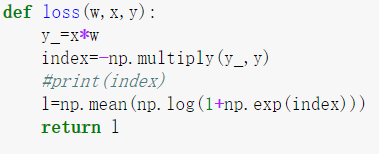
(8)Repeate step 4 to 6 for several times, and drawing graph of LNAG，LRMSprop，LAdaDelta and LAdam  and with the number of iterations.

1. **Code:**

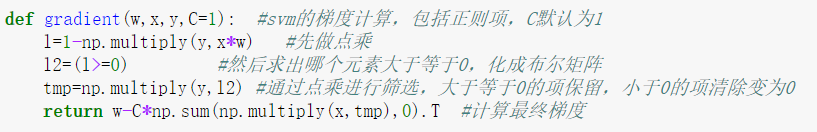
**Logistic regression gradient:**



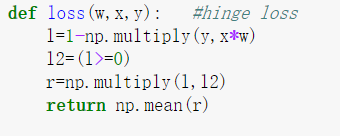
**Logistic regression loss function:**



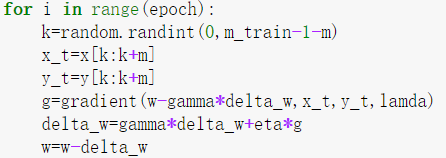
**Linear Classification gradient:**



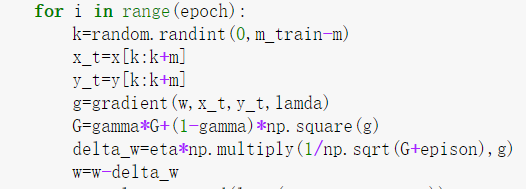
**Linear Classification loss function:**



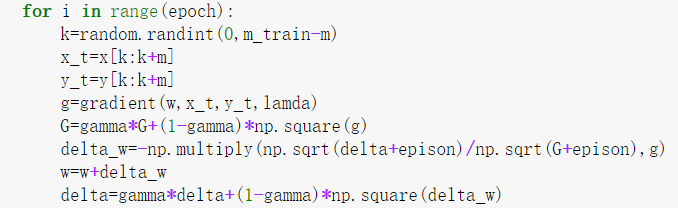
**NAG main code:**



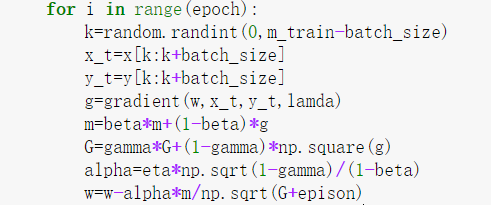
**RMSprop main code:**



**AdaDelta main code:**



**Adam main code:**



1. **The initialization method of model parameters:**

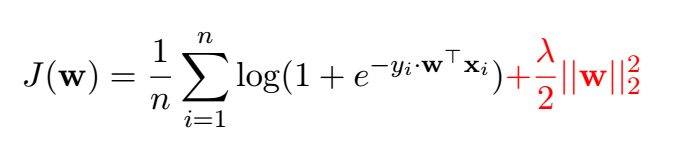
**Logistic Regression:**random numbers or normal distribution.

**Linear Classification:**random numbers or normal distribution.

1. **The selected loss function and its derivatives:**

**Logistic Regression:**

**Loss function:**

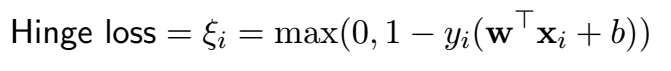


**Derivatives:**

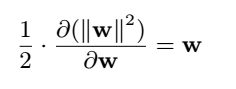
****

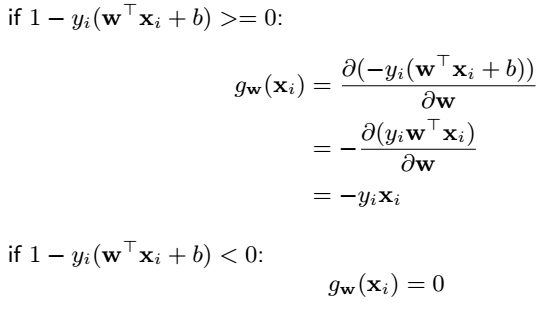
**Linear Classification:**

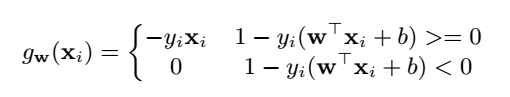
**Loss function:**

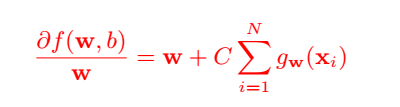


**Derivatives:**









1. **Experimental results and curve:**

**Logistic Regression:**

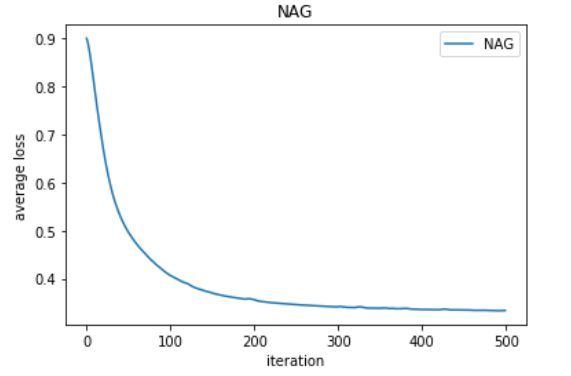
**(1)NAG**

## Hyper-parameter selection:

η=0.05 λ=0.001 γ=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.61%

## Loss curve:



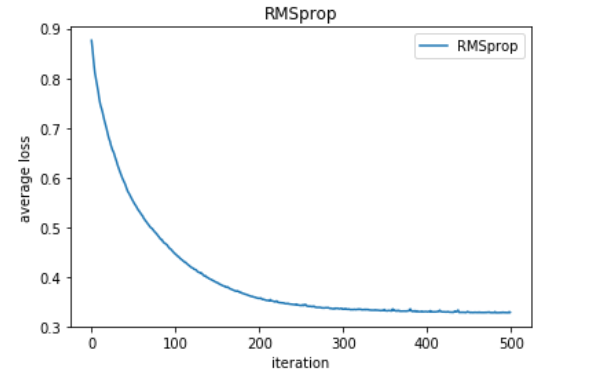
1. **RMSprop**

## Hyper-parameter selection:

η=0.01 λ=0.001 γ=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.97%

## Loss curve:



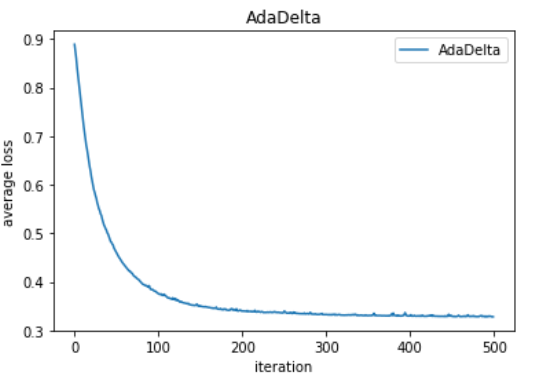
1. **AdaDelta**

## Hyper-parameter selection:

λ=0.001 γ=0.4 batch\_size=256 epoch=500

## Predicted Results (Best Results):85.03%

## Loss curve:



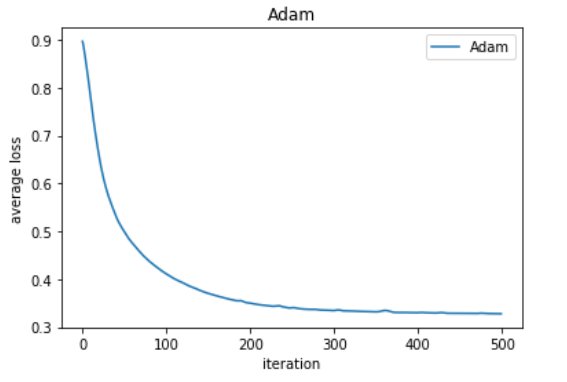
1. **Adam**

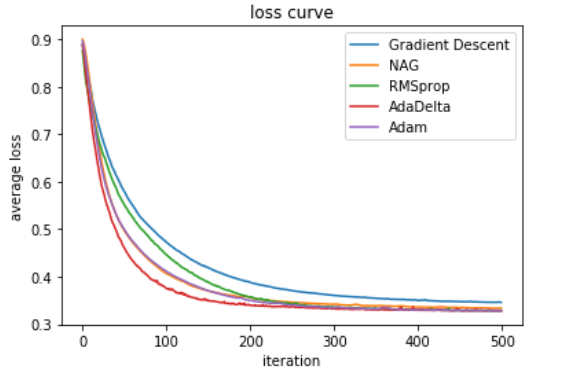
## Hyper-parameter selection:

η=0.005 λ=0.001 γ=0.99 β=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.96%

## Loss curve:





**Linear Classification:**

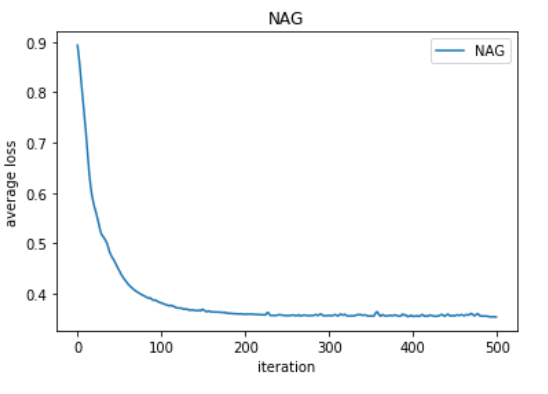
**(1)NAG**

## Hyper-parameter selection:

η=0.0003 C=1 γ=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.61%

## Loss curve:



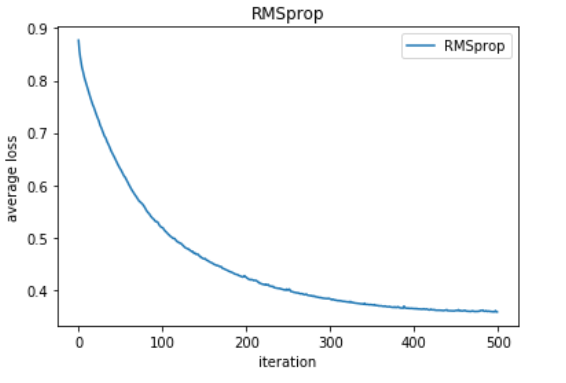
**(2)RMSprop**

## Hyper-parameter selection:

η=0.005 C=1 γ=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.78%

## Loss curve:



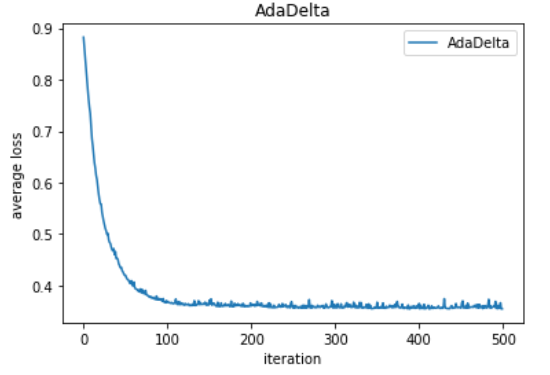
**(3)AdaDelta**

## Hyper-parameter selection:

C=0.9 γ=0.3 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.95%

## Loss curve:



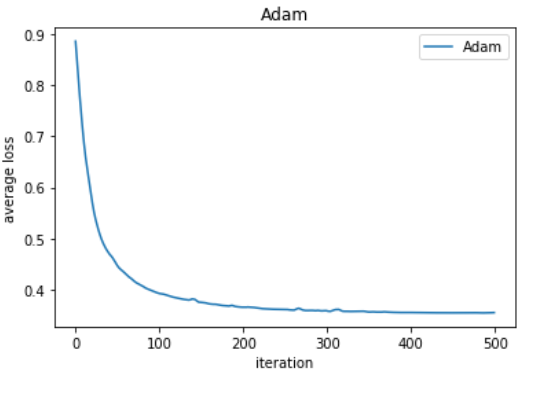
**(4)Adam**

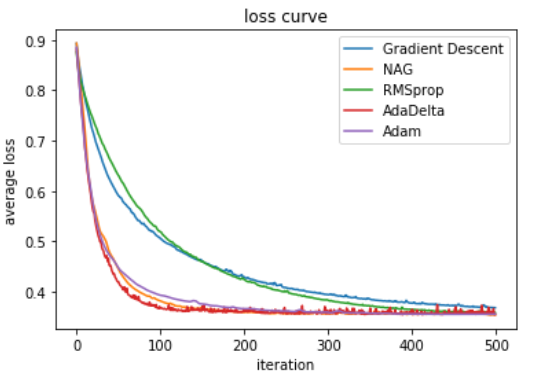
## Hyper-parameter selection:

η=0.01 C=1 γ=0.999 β=0.9 batch\_size=256 epoch=500

## Predicted Results (Best Results):84.78%

## Loss curve:





1. **Results analysis:**

In this experiment, we compare logistic regression and linear classification. We can find in the loss curve that the loss of the logistic regression is more stable. And their accuracy is similar. We also compare different stochastic gradient descent in this experiment. We can find that AdaDelta converge fastest. Adam and the NAG are similar in the convergence rate. RMSprop converge a little faster than the mini-batch gradient descent.

1. **Similarities and differences between logistic regression and linear classification：**

**Similarities:** They are both used in classification problem. When the threshold in linear classification is 0 and the threshold in logistic regression is 0.5, logistic regression and linear classification have similar meanings.

**Differences:**The loss and goal functions of them are different. Logistic regression can be used to solve multiclass classification problem while svm is hard to do that.

1. **Summary:**

In this experiment, we use the logistic regression and linear classification to solve the classification problem. Also, we try different stochastic gradient descent and compare them. The stochastic gradient descent can solve the memory error and speed up the training process but it will make the loss unstable. Different algorithms have different convergence rates. Adadelta converge fastest and it doesn’t require the learning rate but its loss is unstable. Adam and NAG also run fast. And RMSprop is a little faster than the gradient descent.